Web Kit

Internet Technologies

Session 420

Web Kit
Internet Technologies

Richard Williamson
Safari and Web Kit Engineer

Introduction

N
NE
E
SE
S
SW
W
NW

Web Kit

What Is Web Kit?

- A framework
- Embeds web content
  - HTML
  - Images
  - Plugins: Flash, QuickTime, etc.
- Encapsulates internals of open source engine

Demo

Design Goals of Web Kit

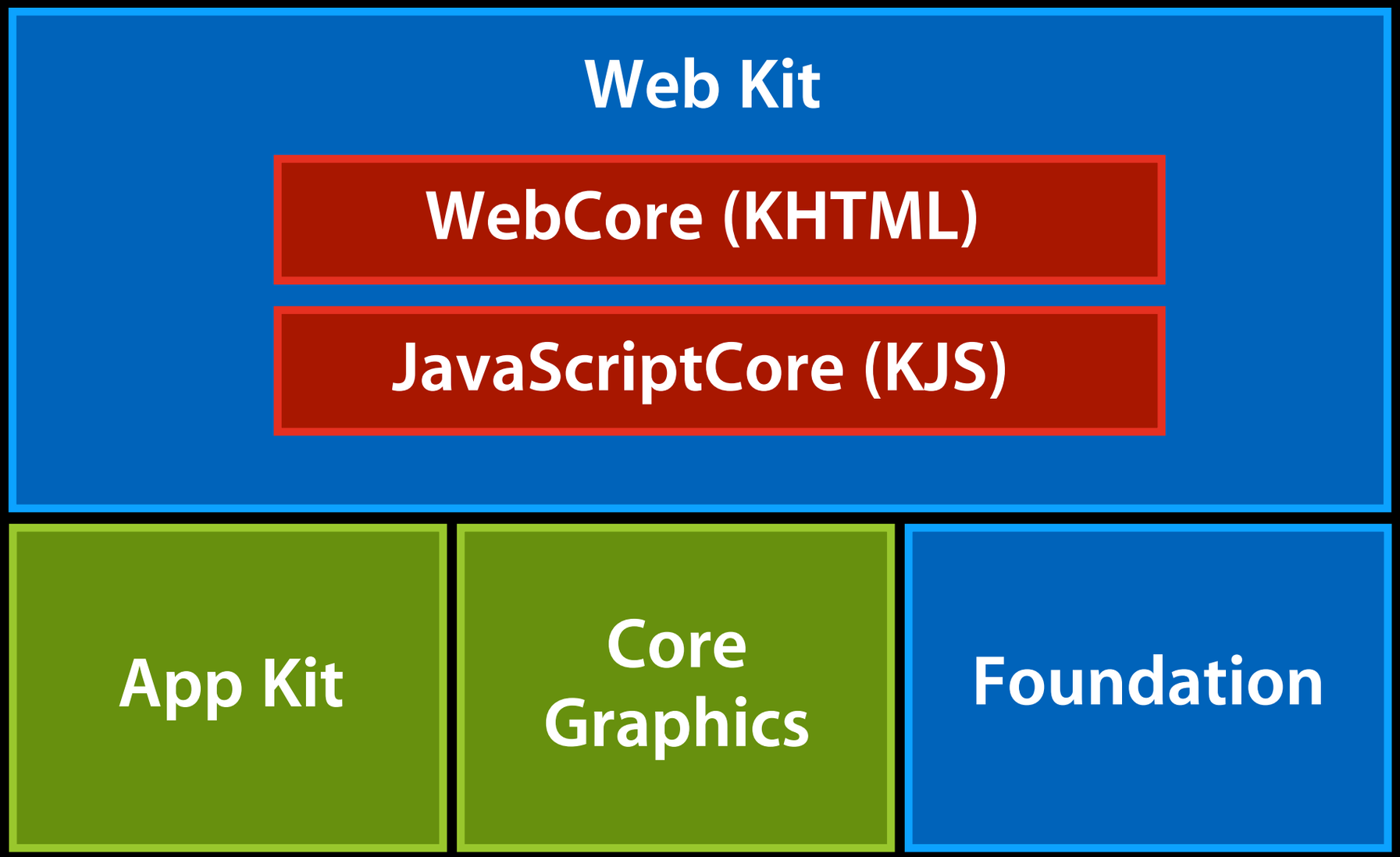
- Support a world-class browser: Safari
- Embeddable HTML/Web technology
- High performance
  - Small foot-print
  - Fast loading
  - Fast rendering
- Excellent standards support
- Simple and flexible API

Design Goals of Web Kit

- Support a world-class browser: Safari
- Embeddable HTML/Web technology
- High performance
  - Small foot-print
  - Fast loading
  - Fast rendering
- Excellent standards support
- Simple and flexible API

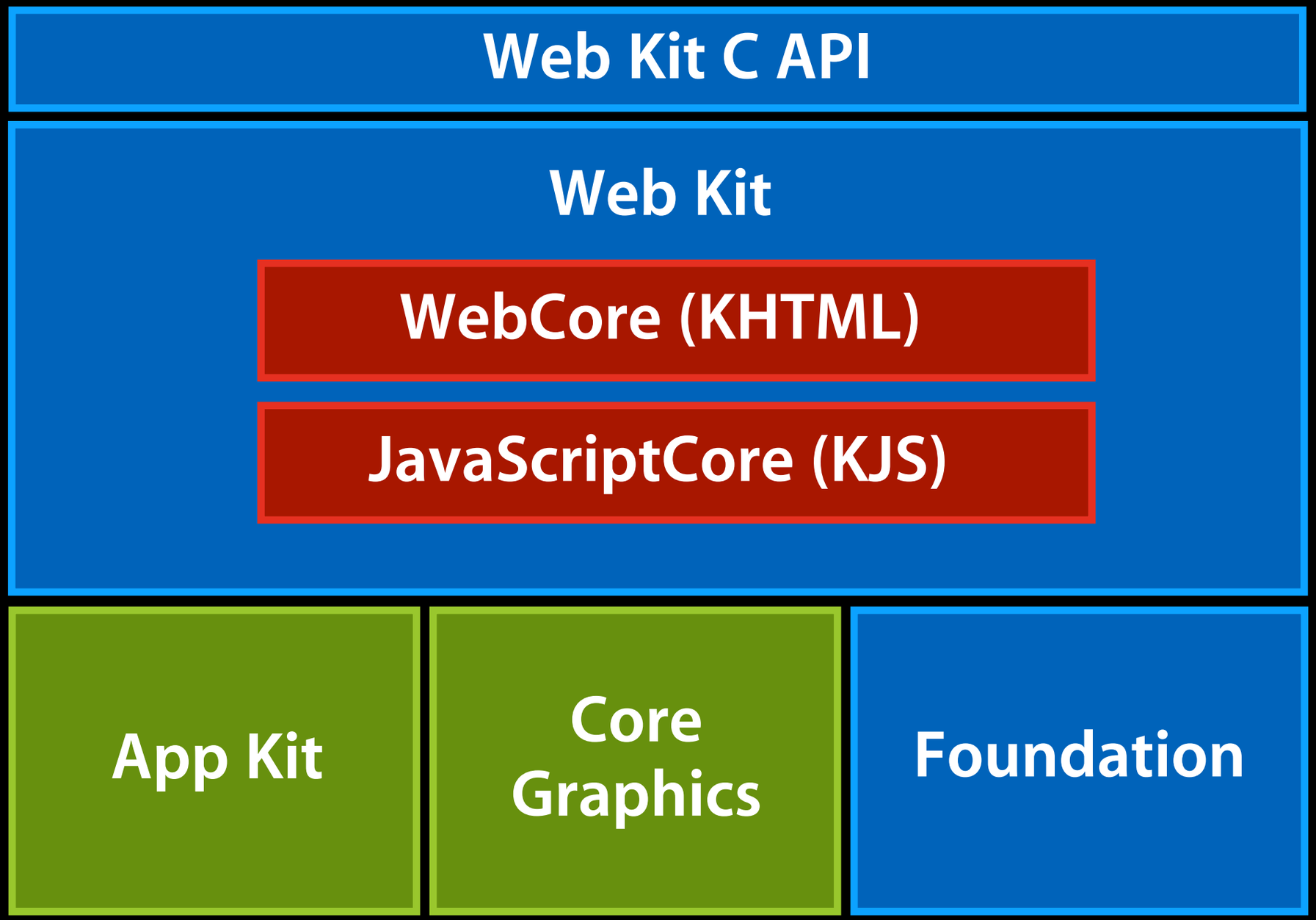
Architecture

Inside Web Kit



Architecture

Inside Web Kit



Web Kit and KHTML

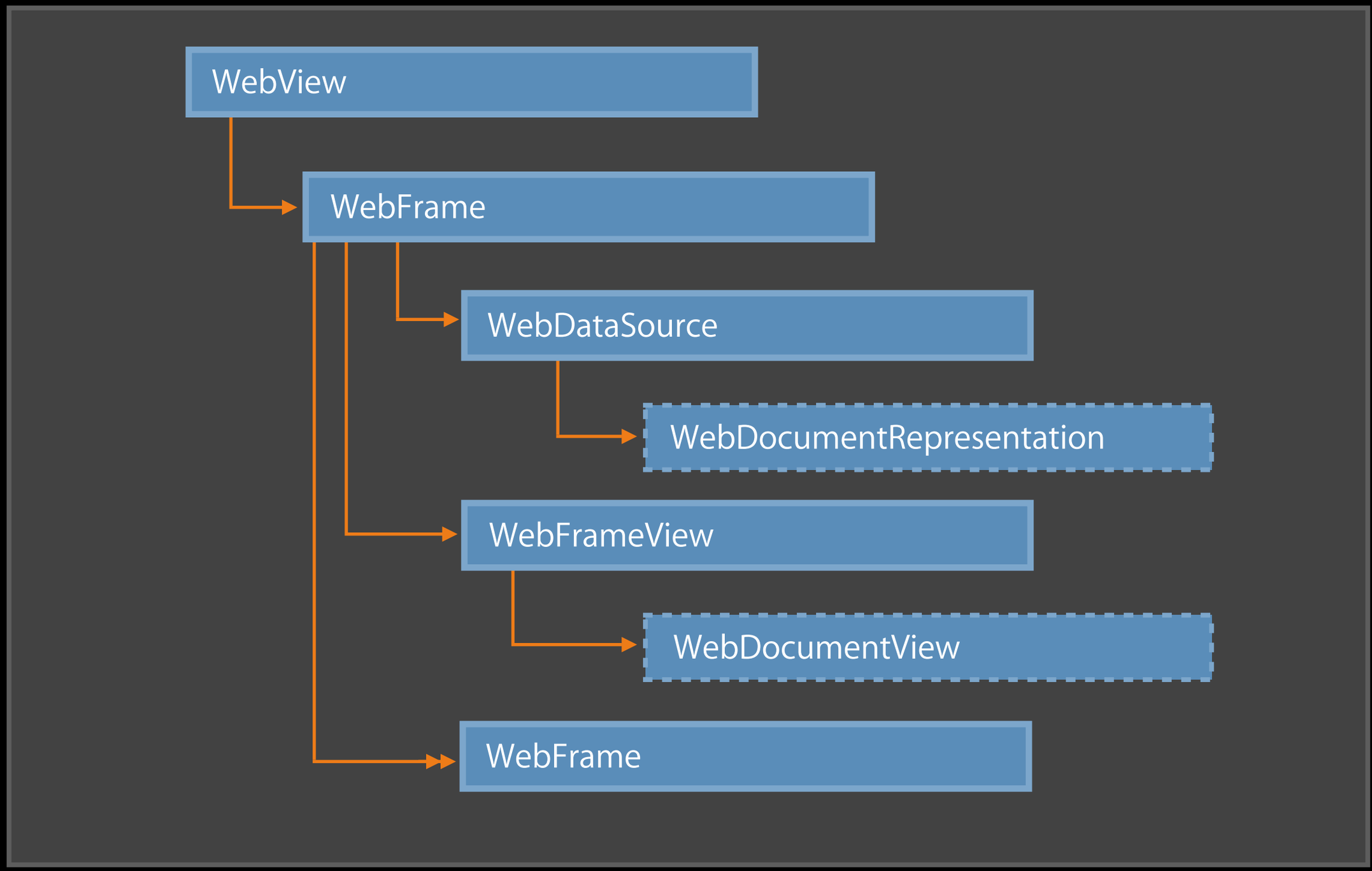
- Web Kit is based on KHTML
- WebCore and JavaScriptCore
  - Open source
  - Include Apple's work on KHTML
- WebCore does **NOT** have a public API

Use the Web Kit API

Key Web Kit Concepts

- Views
- Frames
- Loading data
- Delegates

Classes
Relationships



Classes

```
<HTML>
<HEAD>
</HEAD>

<BODY>
Hello world!
</BODY>

</HTML>
```

Classes

WebView

```
<HTML>
<HEAD>
</HEAD>

<BODY>
Hello world!
</BODY>

</HTML>
```

Classes

WebView

```
<HTML>
<FRAMESET>

<FRAME SRC="a.html">
<FRAME SRC="b.html">

</FRAMESET>
</HTML>
```

Classes

WebView

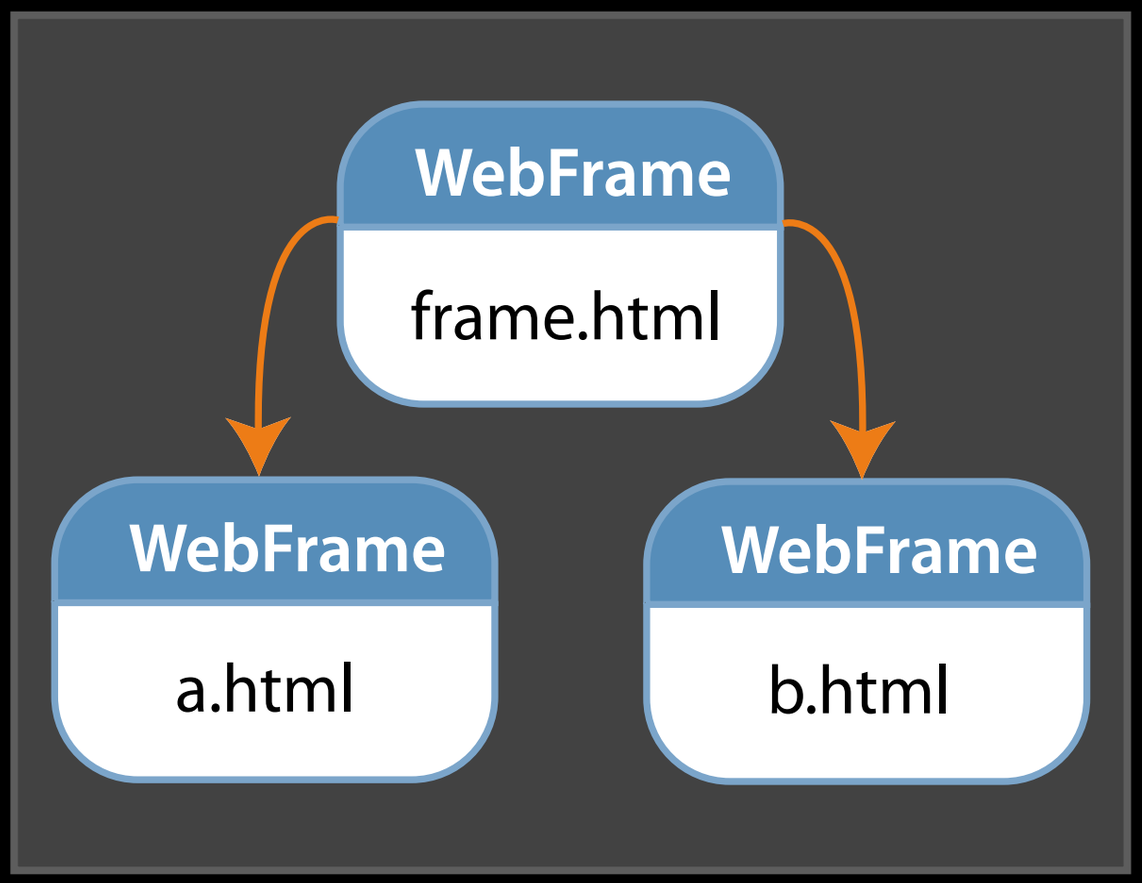
WebFrame

```
<HTML>
<FRAMESET>

<FRAME SRC="a.html">
<FRAME SRC="b.html">

</FRAMESET>
</HTML>
```

Classes

WebView

WebFrame




```
<HTML>
<FRAMESET>

<FRAME SRC="a.html">
<FRAME SRC="b.html">

</FRAMESET>
</HTML>
```

Classes

WebView

WebFrame

WebDataSource

WebFrameView

```
<HTML>
<FRAMESET>

<FRAME SRC="a.html">
<FRAME SRC="b.html">

</FRAMESET>
</HTML>
```

Classes

WebView

WebFrame

WebDataSource

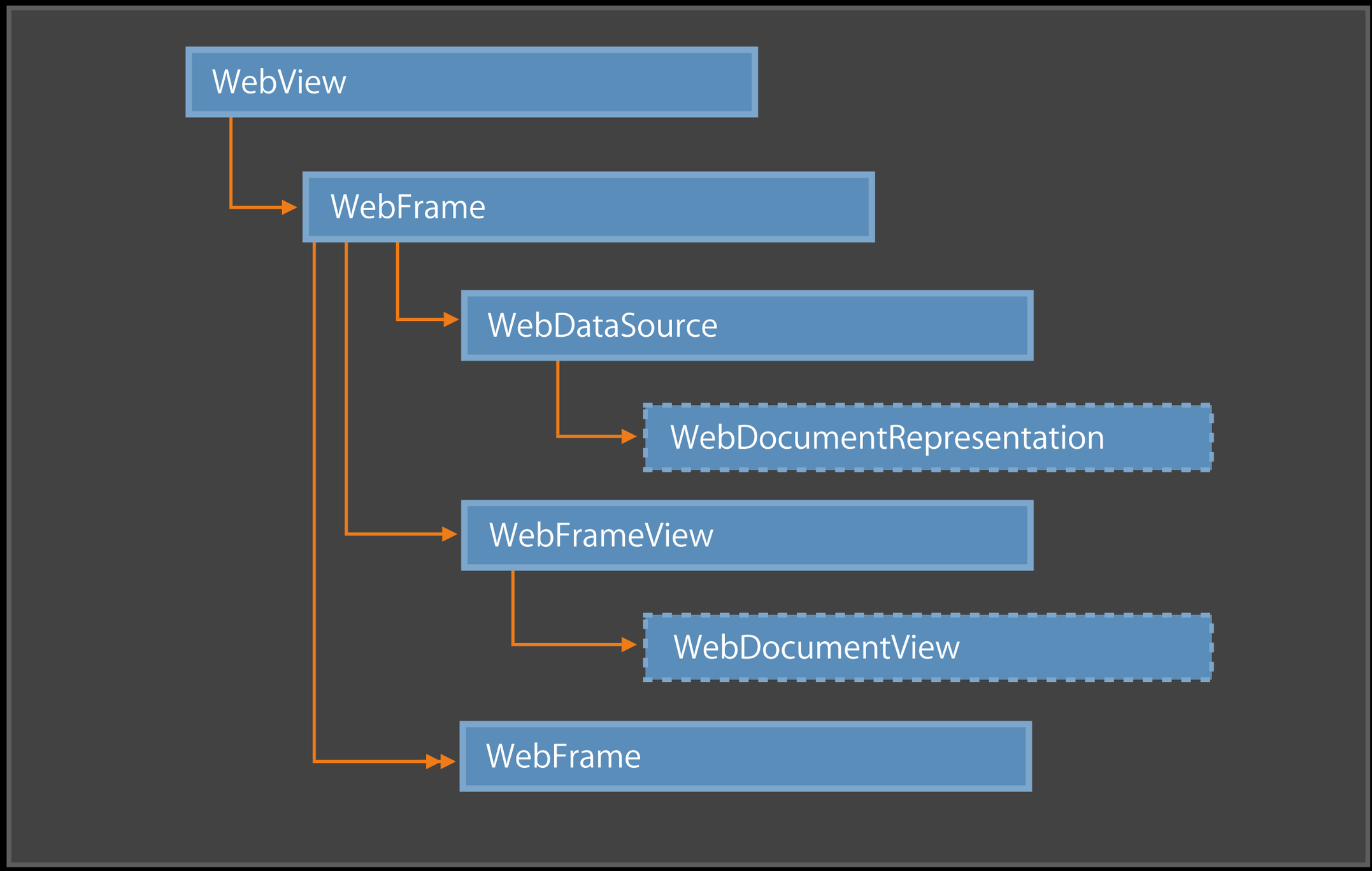
WebDocumentRepresentation

WebFrameView

WebDocumentView

```
<HTML>
<FRAMESET>

<FRAME SRC="a.html">
<FRAME SRC="b.jpg">

</FRAMESET>
</HTML>
```

Classes
Relationships



Loading…

- Uses new Foundation URL loading system
- NSURLRequest
  - Represents a request for data
- NSURLResponse
  - Represents a response for a request
- Review Session 418: Advanced Foundation URL APIs
  - Available on DVD

Loading…

- Initiated on WebFrames
- Content types
  - NSURLRequest
  - NSString
  - NSData
- Progress can be monitored

Loading…

- Web pages include a main document and resources
- Resource types
  - Images
  - CSS
  - JavaScript
  - HTML
- Resources load automatically
- Progress can be monitored

Code Example
Create A WebView

```
// Create a WebView.
WebView *webView = [[WebView alloc]
          initWithFrame:NSMakeRect(0,0,640,480)];

// Add the WebView to the view hierarchy.
[contentView addSubview:webView];
```

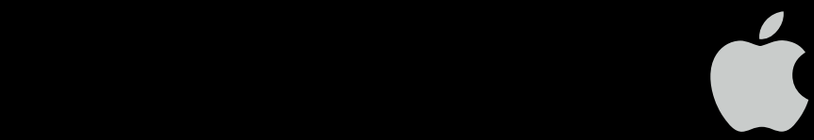
Code Example

Load A URL

```
// Create a WebView.
WebView *webView = [[WebView alloc]
          initWithFrame:NSMakeRect(0,0,640,480)];

// Add the WebView to the view hierarchy.
[contentView addSubview:webView];

// Get the main frame for the WebView.
WebFrame *mainFrame = [webView mainFrame];

// Load a URL into the main frame.
NSURLRequest *aReq = [NSURLRequest requestWithURL:aURL];
[mainFrame loadRequest: aReq];
```

Code Example
Load A String

```
// Create a WebView.
WebView *webView = [[WebView alloc]
          initWithFrame:NSMakeRect(0,0,640,480)];

// Add the WebView to the view hierarchy.
[contentView addSubview:webView];

// Get the main frame for the WebView.
WebFrame *mainFrame = [webView mainFrame];

// Create an HTML snippet.
NSString *HTMLString = @"<b>Hello world!</b>";

// Load a URL into the main frame.
[mainFrame loadHTMLString:HTMLString baseURL:nil];
```



Code Example
Log Document Source

```
// Get the main frame for the WebView.
WebFrame *mainFrame = [webView mainFrame];

// Get the document representation of the main frame.
id<WebDocumentRepresentation> doc;
doc = [[mainFrame dataSource] representation];

// If we can get a string representation, log it.
if ([doc canProvideDocumentSource])
    NSLog (@"%@", [doc documentSource]);
```

Interacting With an HTML Document

- JavaScript can be executed in document's context
- A typical approach
  - Define a JavaScript function
  - Invoke from Web Kit
- Document Object Model (DOM) can be accessed from JavaScript

Code Example

JavaScript Interaction

```
<script type="text/javascript">
/*
    Shrink the image dimensions
*/
function zoomImageOut(imageID)
{
    var anImage = document.getElementById(imageID);
    anImage.width = parseInt(anImage.width) / 1.5;
}
</script>

...

<img id="myImage" src="myImage.jpg">
```

Code Example

JavaScript Interaction

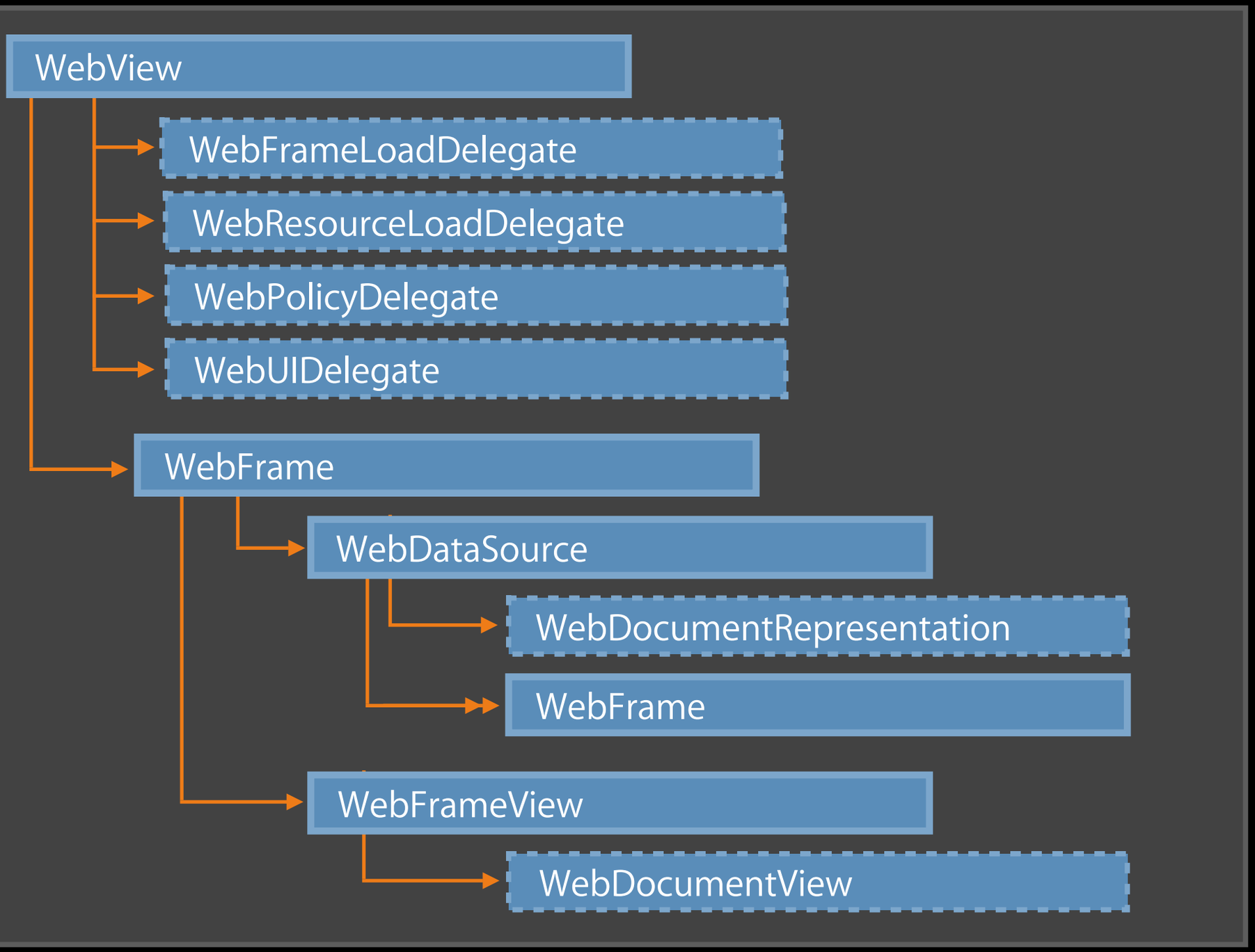
```
// Create a JavaScript snippet.
NSString *JSString;
JSString = [NSString stringWithFormat:
                    @"zoomImageOut(%@)", myImageName];

// Execute JavaScript in WebView's document context.
[webView stringByEvaluatingJavaScriptFromString:JSString];
```

Demo

Delegates

- Web Kit behavior controlled by delegates
- Delegates respond to Web Kit informal protocols
- Methods **optionally** implemented
- Delegates managed by WebView

Delegates
Relationships

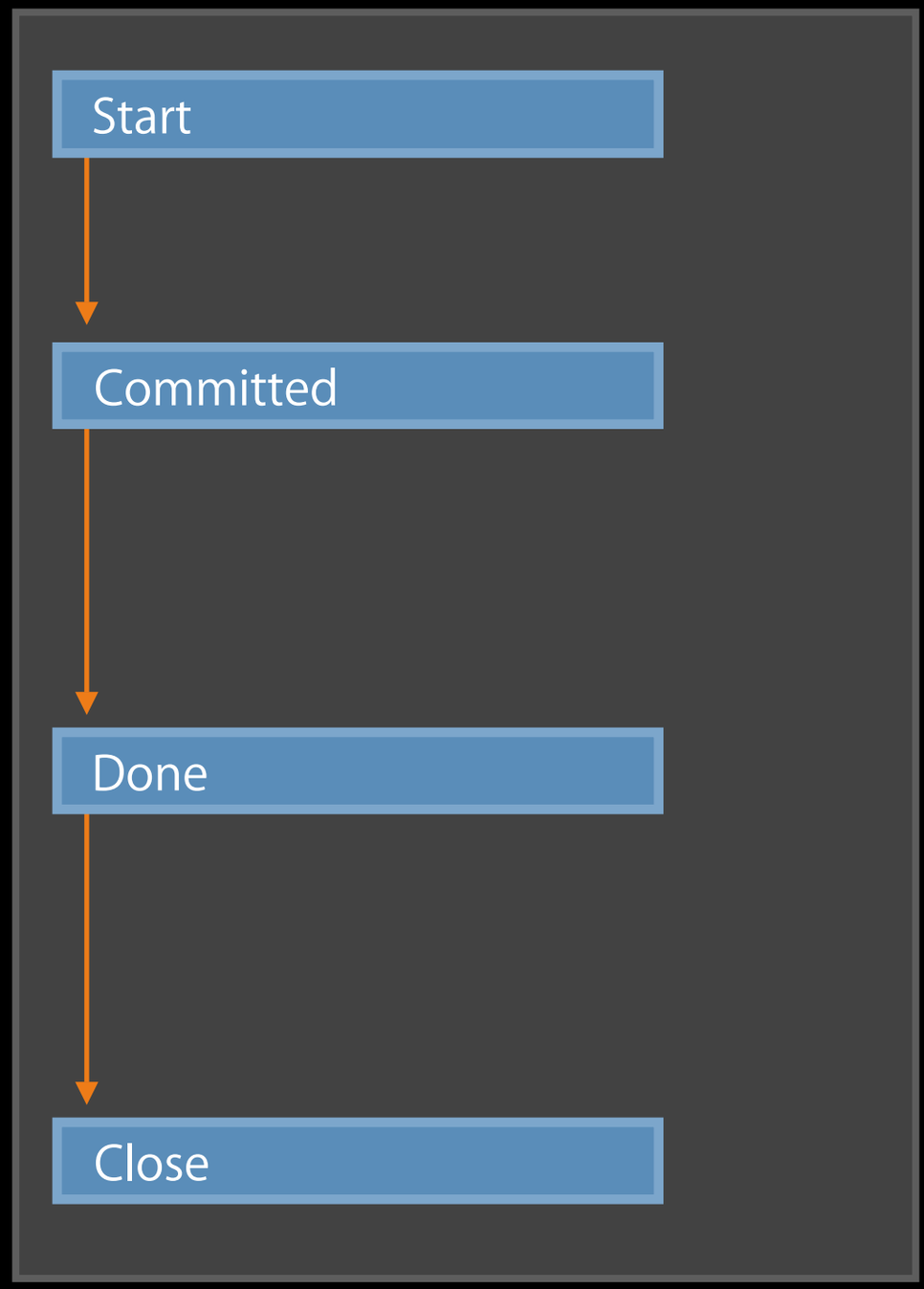


Delegates

- WebFrameLoadDelegate
- WebResourceLoadDelegate
- WebPolicyDelegate
- WebUIDelegate

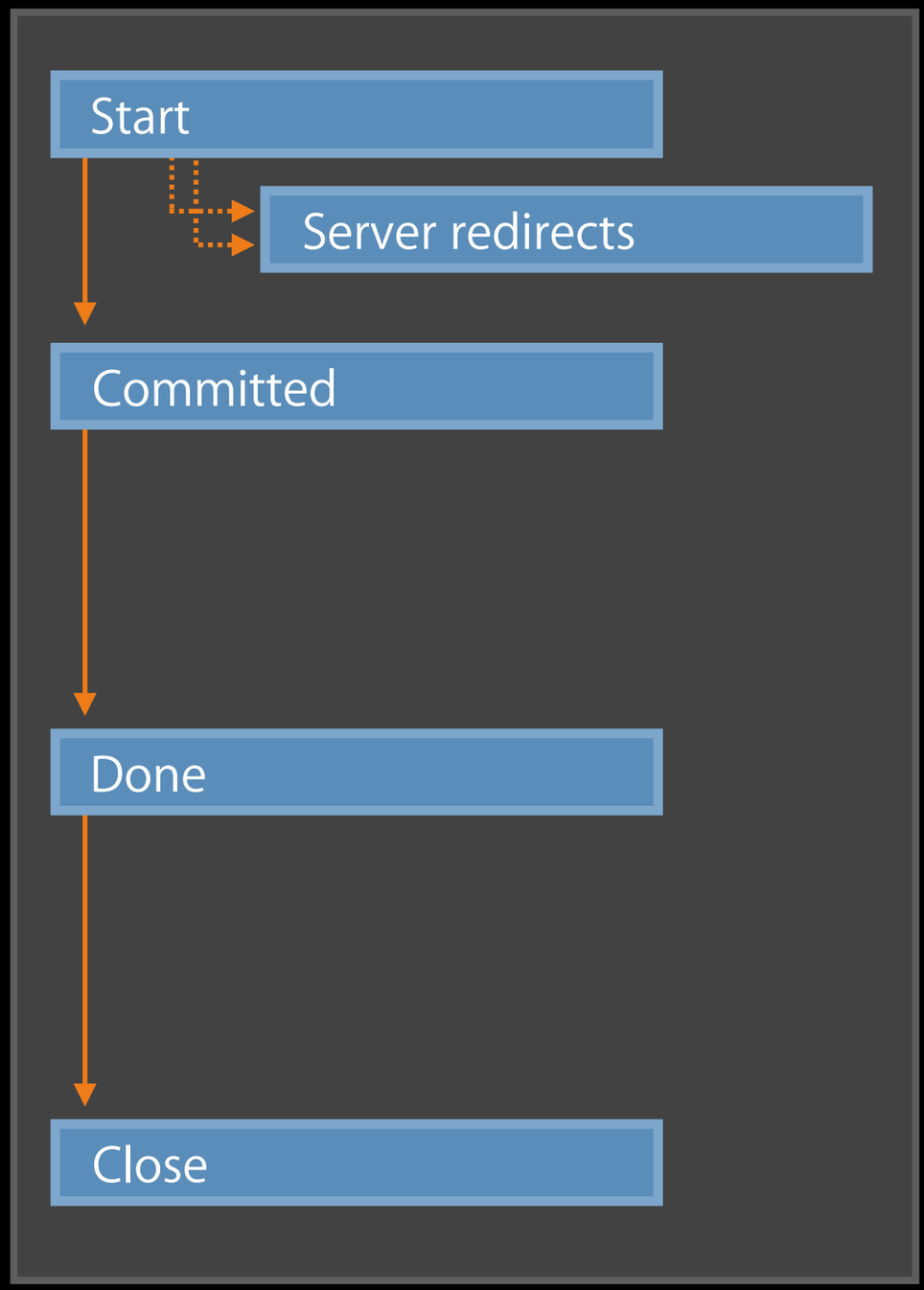
WebFrameLoadDelegate

- Tracks load progress of a document
  - Start
  - Committed
  - Done
  - Close



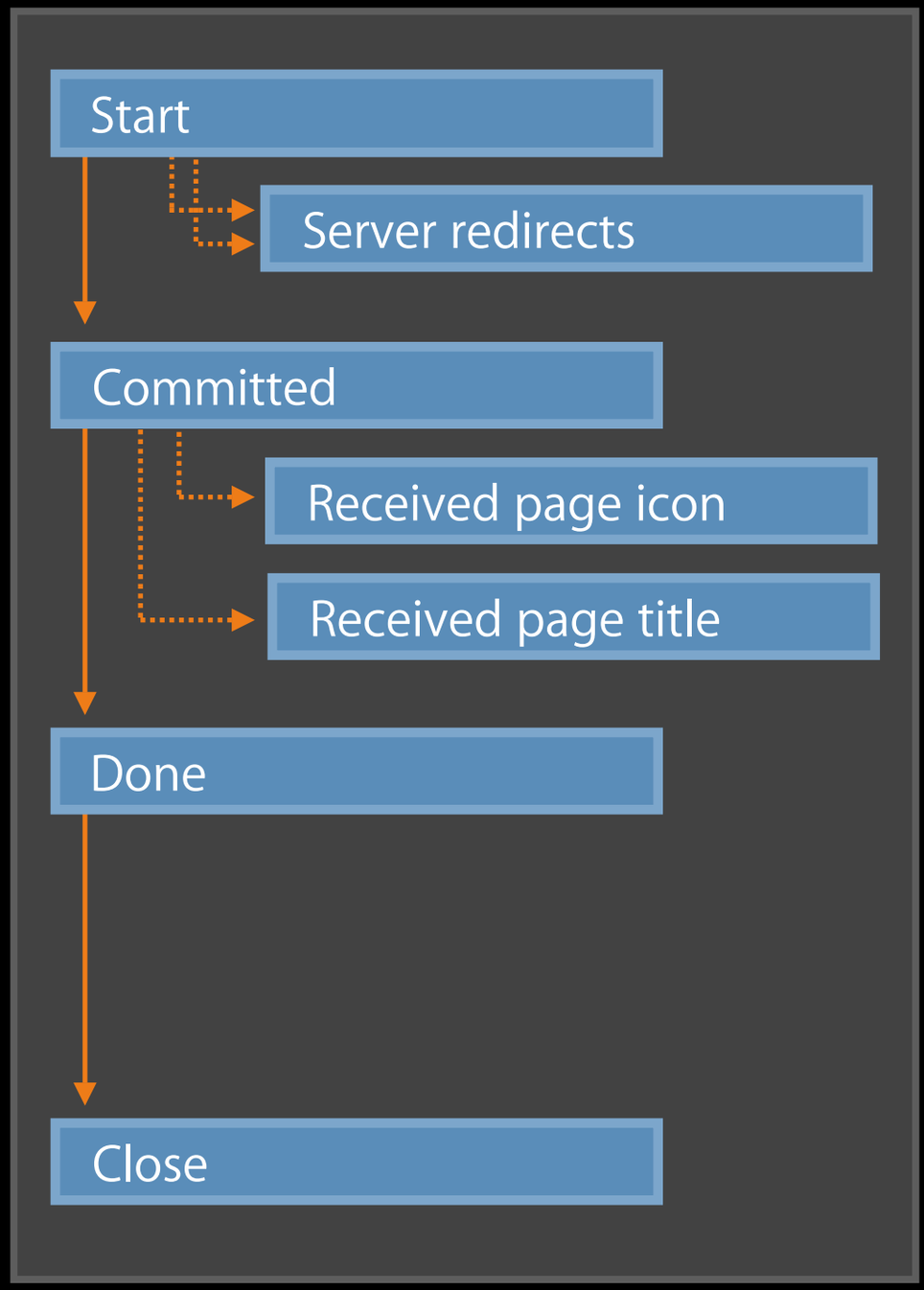
WebFrameLoadDelegate

- Tracks changes to the document shown in a WebView
  - Start
  - Server redirects
  - Committed
  - Done
  - Close



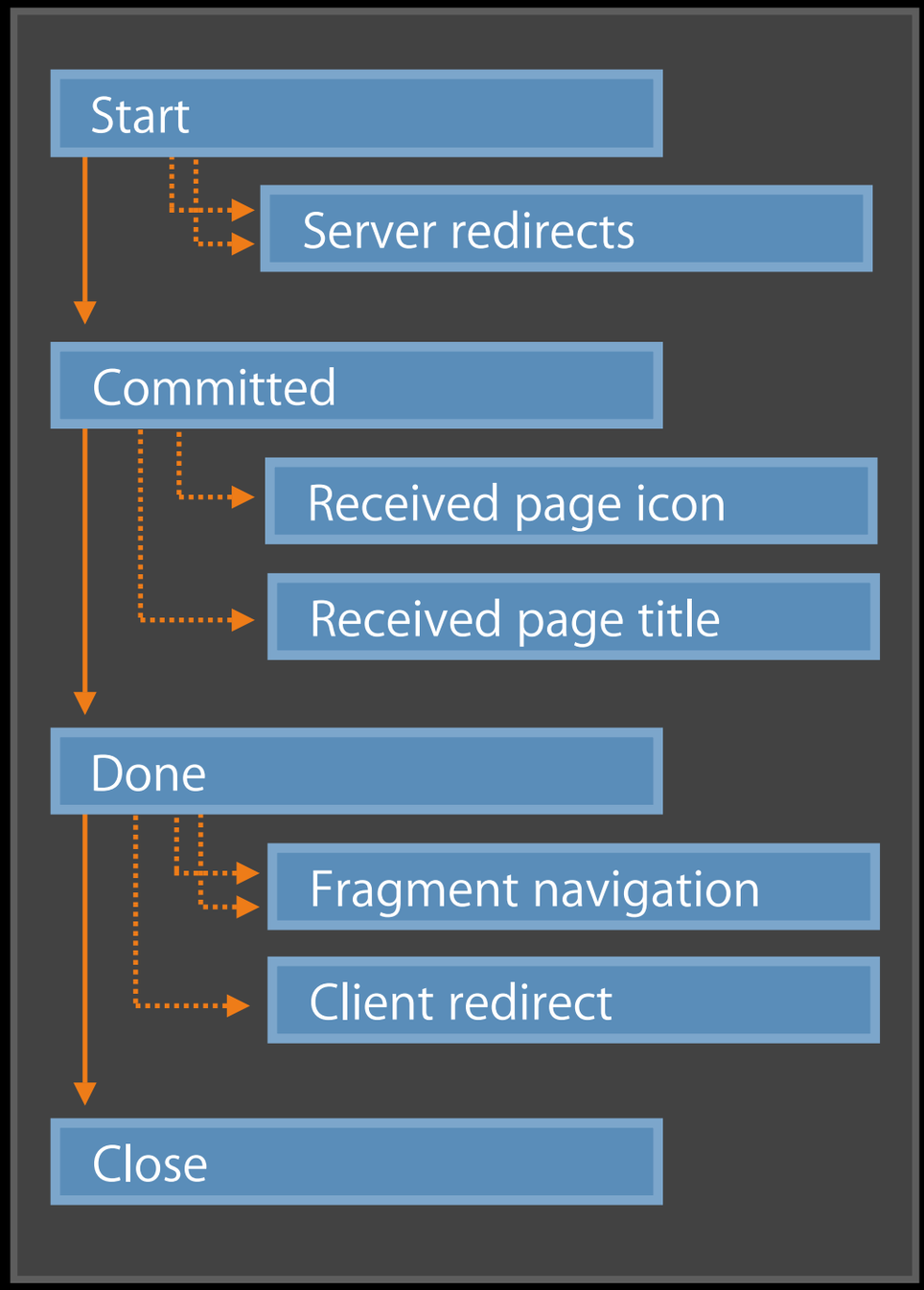
WebFrameLoadDelegate

- Tracks changes to the document shown in a WebView
  - Start
  - Server redirects
  - Committed
  - Received page icon
  - Received page title
  - Done
  - Close



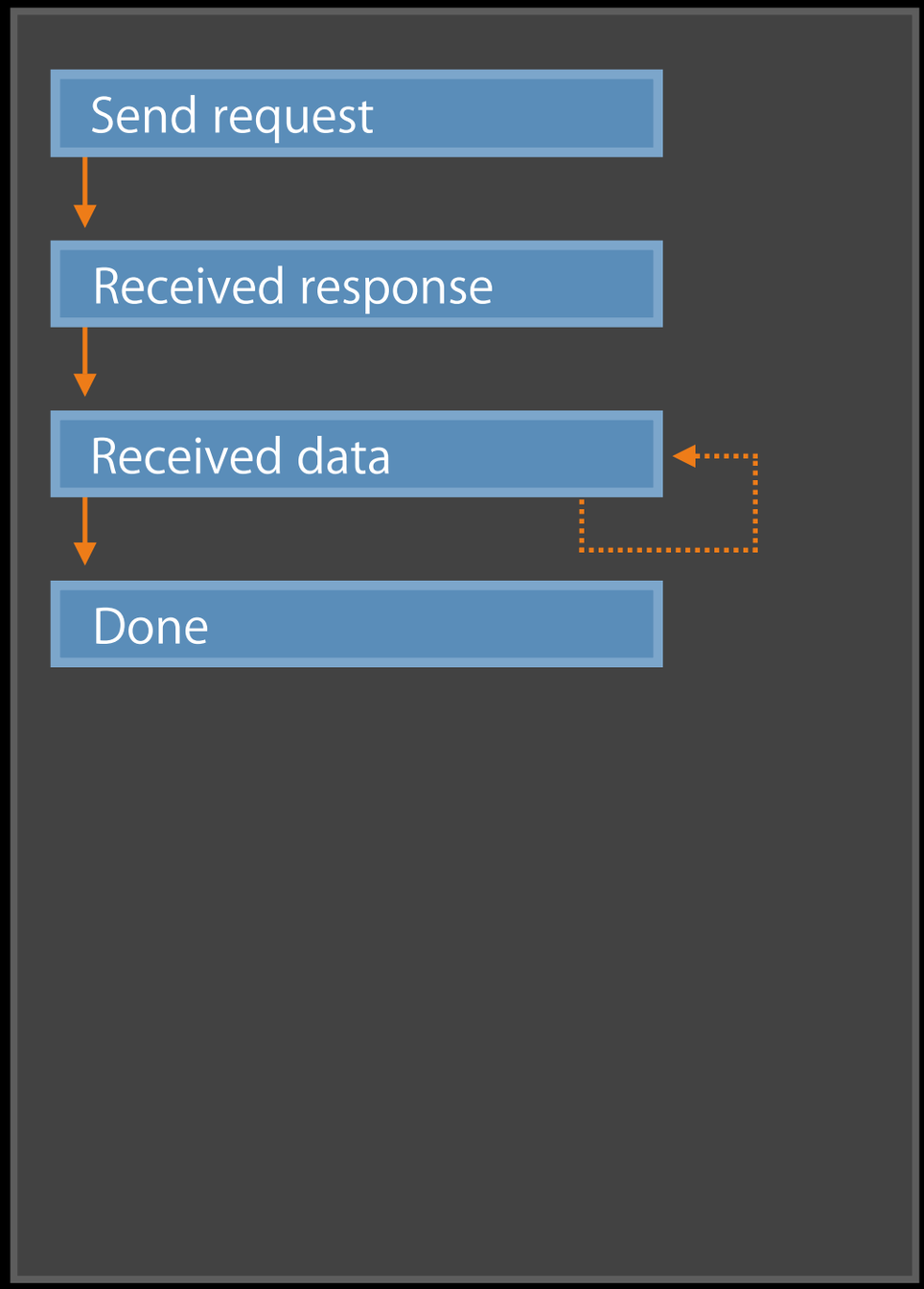
WebFrameLoadDelegate

- Tracks changes to the document shown in a WebView
  - Start
  - Server redirects
  - Committed
  - Received page icon
  - Received page title
  - Done
  - Fragment navigation
  - Client redirects
  - Close



WebResourceLoadDelegate

- Tracks loading of resources used by a page
  - Send request
  - Received response
  - Received data
  - Done



WebPolicyDelegate

- Decides how navigation actions are handled
  - Can be asynchronous
  - Each policy choice has one of three possible responses
    1. Use
    2. Ignore
    3. Download

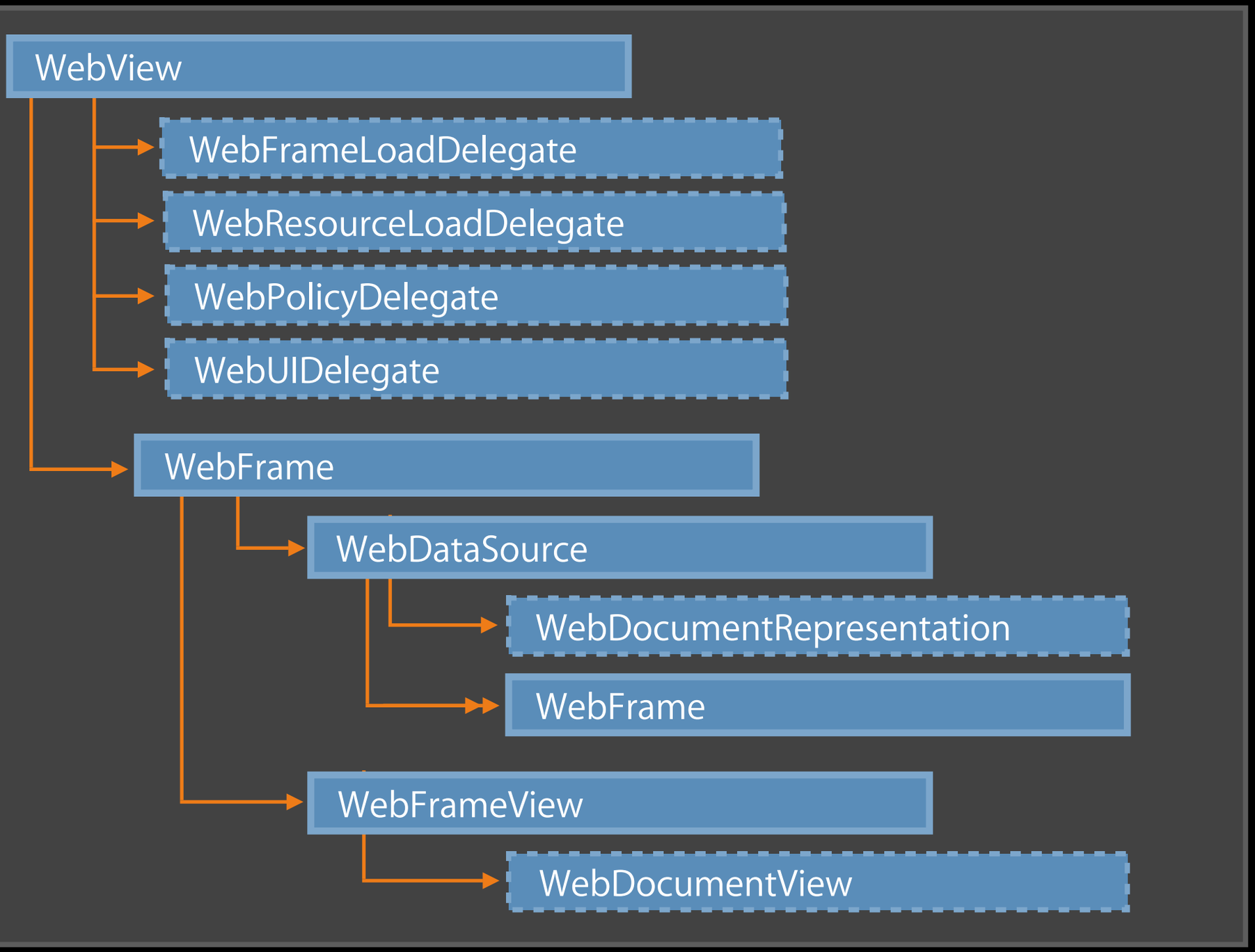
WebPolicyDelegate

- Consulted at least three times during a load
  1. Open window?
  2. Handle request?
  3. Handle MIME type?

WebUIDelegate

- Manages window-related activity
- Shows JavaScript status text
- Presents JavaScript alerts
- Manages window *chrome*
- Customizes context menus

Delegates
Relationships



Demo

Web Kit C API

- Can be used in Carbon applications
- WebView is available as an HIView
- Objective-C API is available if you need it

Web Kit ‘C’ API

Code Snippet

```
// Create a WebView.
HIWebViewCreate( &webView );

// Get contentView of window.
HIViewFindByID( HIViewGetRoot( window ),
                kHIViewWindowContentID,
                &contentView );

// Set webViews bounds to contentViews bounds.
HIViewGetBounds( contentView, &bounds );
HIViewSetFrame( webView, &bounds );

// Add the webView to the view hierarchy.
GetRootControl( window, &root );
HIViewAddSubview( root, webView );
```

Demo

Web Kit SDK

- Works on OS X 10.2+
- **Requires** Safari 1.0 installation
- Several examples, including more comprehensive versions of the demos shown today
- Conceptual and reference documentation
- Available now from ADC

For More Information

Reference Library

Documentation

Displaying Web Content

ADC Home > Documentation > Cocoa > User Experience >

Web Kit Reference for Objective-C

ADC Home > Documentation > Cocoa >

Web Kit Reference for C

ADC Home > Documentation > Carbon > HIToolbox >

URL Loading System

ADC Home > Documentation > Cocoa > Networking >

Reference Library

Sample Code

Web Kit SDK Sample Code

Available in the /Developer/Examples/WebKit folder

Reference Library

Other Resources

“HTML and XHTML: The Definitive Guide”
Published by O’Reilly & Associates

“Cascading Style Sheets: The Definitive Guide”
Published by O’Reilly & Associates

“JavaScript: The Definitive Guide”
Published by O’Reilly & Associates

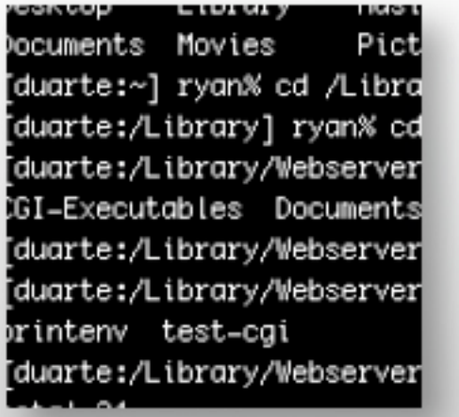
World Wide Web Consortium
www.w3.org

Who to Contact

John Geleynse
Manager, Software Evangelism
geleynse@apple.com

http://developer.apple.com/wwdc2003/urls.html

Q&A



http://developer.apple.com/wwdc2003/urls.html

WWDC 2003

Worldwide Developers Conference